Configuring Authentication and Authorization

**User:**

In the OpenShift Container Platform architecture, users are entities that interact with the API server. The user resource represents an actor within the system. Assign permissions by adding roles to the user directly or to the groups of which the user is a member.

**Identity**

The identity resource keeps a record of successful authentication attempts from a specific user and identity provider. Any data concerning the source of the authentication is stored on the identity. Only a single user resource is associated with an identity resource.

**Service Account**

In OpenShift, applications can communicate with the API independently when user credentials cannot be acquired. To preserve the integrity of a regular user's credentials, credentials are not shared and service accounts are used instead. Service accounts enable you to control API access without the need to borrow a regular user's credentials

**Role**

A role defines a set of permissions that enables a user to perform API operations over one or more resource types. You grant permissions to users, groups, and service accounts by assigning roles to them.

### Authenticating API Requests:

Authentication and authorization are the two security layers responsible for enabling user interaction with the cluste.

The OpenShift API has two methods for authenticating requests:

* OAuth Access Tokens
* X.509 Client Certificates

#### Introducing the Authentication Operator:

The OpenShift Container Platform provides the Authentication operator, which runs an OAuth server.

**HTPasswd**

Validates user names and passwords against a secret that stores credentials generated using the htpasswd command.

A newly-installed OpenShift cluster provides two ways to authenticate API requests with cluster administrator privileges:

* Use the kubeconfig file, which embeds an X.509 client certificate that never expires.
* Authenticate as the kubeadmin virtual user. Successful authentication grants an OAuth access token

#### Authenticating Using the X.509 Certificate

During installation, the OpenShift installer creates a unique **kubeconfig** file in the auth directory. The **kubeconfig** file contains specific details and parameters used by the CLI to connect a client to the correct API server, including an X.509 certificate.

To use the**kubeconfig** file to authenticate oc commands, you must copy the file to your workstation and set the absolute or relative path to the **KUBECONFIG** environment variable:

[user@host ~]$ export KUBECONFIG=/home/user/auth/kubeconfig

[user@host ~]$ oc get nodes

As an alternative, you can use the **--kubeconfig**option of the oc command.

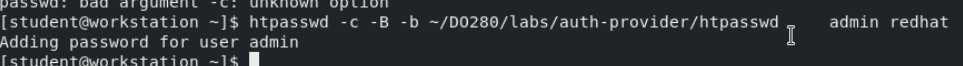
[user@host ~]$ oc --kubeconfig /home/user/auth/kubeconfig get nodes

#### Deleting the Virtual User

After you define an identity provider, create a new user, and assign that user the cluster-admin role, you can remove the kubeadmin user credentials to improve cluster security.

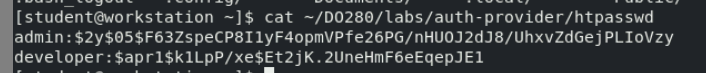
[user@host ~]$ oc delete secret kubeadmin -n kube-system

Create an HTPasswd authentication file named htpasswd:



Add the developer user with a password of developer to the ~/DO280/labs/auth-provider/htpasswd file.





OpenShift and create a secret that contains the HTPasswd users file

Log in to the cluster as the kubeadmin user.